Capítulo 16: Representabilidad de las funciones recursivas *Computability and Logic*, Quinta edición

Para refrescar la memoria

- El *lenguaje de la aritmética* L^* tiene como símbolos no lógicos a la constante cero $\mathbf{0}$, al predicado diádico menor $<$, al símbolo de función monádico sucesor $'$ y a los símbolos de función diádicos producto $\cdot$ y suma $+$.
- La *interpretación estándar del lenguaje de la aritmética* $\mathcal{N}^*$ tiene como dominio a $\mathbb{N}$. La denotacion $\mathbf{0}^{\mathcal{N}^*}$ de $\mathbf{0}$ es el número cero, la denotación $<^{\mathcal{N}^*}$ de $<$ es la relación de ser menor, la denotación $'^{\mathcal{N}^*}$ de $'$ es la función sucesor y las denotaciones $+^{\mathcal{N}^*}$ y $\cdot^{\mathcal{N}^*}$ para $+$ y $\cdot$ son las funciones de suma y producto, respectivamente.
- La *aritmética verdadera* es el conjunto Γ de todas las oraciones del lenguaje de la aritmética que son verdaderas en la interpretación estándar.

Definibilidad aritmética

- El objetivo principal del capítulo es mostrar que la aritmética puede hablar acrca de todas las funciones y conjuntos recursivos.
- Decimos que una fórmula $F(x)$ del lenguaje de la aritmética *define aritméticamente* un conjunto S de números naturales si y sólo si:
  - para todo número natural n, Sn si y sólo si $\mathcal{N}^* \models F(\mathbf{n})$.
- Esta noción puede aplicarse también a funciones. Decimos que un fórmula $F(x, y)$ *define aritméticamente* una función monádica f si y sólo si:
  - para cualesquiera números naturales n y m, $f(m) = n$ si y sólo si $\mathcal{N}^* \models F(\mathbf{m}, \mathbf{n})$.
- Esto puede generalizarse de forma obvia a relaciones y funciones k-ádicas.

Ejemplos de funciones aritméticamente definibles

- La función de identidad $id(x)$ puede definirse por medio de la fórmula $x = y$.
- De forma más general, las funciones $id_i^n(x_1, ..., x_n)$ pueden definirse por medio de fórmulas de la forma $x_1 = x_1 \,\&\, ... \,\&\, x_i = y \,\&\, ... \,\&\, x_n = x_n$.
- La función zero $z(x)$ puede definirse por medio de la fórmula $y = \mathbf{0} \,\&\, x = x$.
- La función sucesor x' puede definirse por medio de la fórmula $y = x'$.

- Las funciones suma y producto.
- ¿Qué ocurre con las funciones que no forman parte del lenguaje de la aritmética?
- La función predecesor $pred(x)$ puede definirse por medio de la fórmula $(x = \mathbf{0} \ \& \ y = \mathbf{0}) \vee x = y'$.
- La función diferencia modificada $x_1 \dot{-} x_2$ puede definirse por medio de la fórmula $(x_1 < x_2 \ \& \ y = \mathbf{0}) \vee (x_1 = x_2 + y)$.
- Otros ejemplos pueden encontrarse en el libro (p. 200).

Recursividad y definibilidad

- Nuestro primer resultado importante es el siguiente:
- **Lema 16.6** Todas las funciones recursivas son aritméticamente definibles.

  Esbozo de prueba Ya hemos visto que las funciones básicas *identidad*, *zero* y *sucesor* son aritméticamente definibles. Para probar que todas las funciones recursivas son aritméticamente definibles es suficiente mostrar que:

  1. Si una función se obtiene por *composición* a partir de otras funciones aritméticamente definibles, entonces es aritméticamente definible.
  2. Si una función se obtiene por *recursión primitiva* a partir de otras funciones aritméticamente definibles, entonces es aritméticamente definible.
  3. Si una función se obtiene por *minimización* a partir de otra función aritméticamente definible, entonces es aritméticamente definible.

  Ofreceré solamente la prueba del primer hecho. Sea $h(x) = g(f(x))$, es decir, h se obtiene por composición a partir de las funciones f y g[1]. Por hipótesis, sabemos que f y g son aritméticamente definibles. Es decir, hay fórmulas $\phi_f(x, y)$ y $\phi_g(x, y)$ tales que:

  – para todos los números m y n, se da que $f(m) = n$ si y sólo si $\mathcal{N}^* \vDash \phi_f(\mathbf{m}, \mathbf{n})$.
  – para todos los números m y n, se da que $g(m) = n$ si y sólo si $\mathcal{N}^* \vDash \phi_g(\mathbf{m}, \mathbf{n})$.

  Necesitamos mostrar que hay una fórmula $\phi_h(x, y)$ que define la función h. Sostenemos que dicha fórmula tiene la forma $\exists z(\phi_f(x, z) \ \& \ \phi_g(z, y)$.

  Ahora bien, dado cualquier a, sea $f(a) = b$ y sea $c = h(a) = g(f(a)) = g(b)$. Luego, puesto que $\phi_f(x, y)$ define aritméticamente f y $\phi_g(x, y)$ define

[1] Asumo aquí que h, f y g son funciones monádicas. Por supuesto, la prueba puede generalizarse a funciones de mayor aridad.

aritméticamente g, tenemos $\mathcal{N}^* \vDash \phi_f(\mathbf{a}, \mathbf{b})$ y $\mathcal{N}^* \vDash \phi_g(\mathbf{b}, \mathbf{c})$. Por tanto, tenemos $\mathcal{N}^* \vDash \phi_f(\mathbf{a}, \mathbf{b}) \,\&\, \phi_g(\mathbf{b}, \mathbf{c})$, con lo cual se sigue $\mathcal{N}^* \vDash \exists z(\phi_f(\mathbf{a}, z) \,\&\, \phi_g(z, \mathbf{c})$. Es decir, $\mathcal{N}^* \vDash \phi_h(\mathbf{a}, \mathbf{c})$. Por otra parte, si suponemos que $\mathcal{N}^* \vDash \phi_h(\mathbf{a}, \mathbf{c})$, se sigue que para algún b, $\mathcal{N}^* \vDash \phi_f(\mathbf{a}, \mathbf{b})$ y $\mathcal{N}^* \vDash \phi_g(\mathbf{b}, \mathbf{c})$. Pero dado que $\phi_f(x, y)$ define f, b sólo puede ser $f(a)$, y dado que $\phi_g(x, y)$ define g, c sólo puede ser $g(b) = g(f(a)) = h(a)$. ■

- Las pruebas para el segundo y el tercer hecho pueden encontrarse en el libro (pp. 201-202).
- Si todas las funciones recursivas son aritméticamente definibles, también lo son todos los conjuntos y relaciones recursivas (vía su función característica).

Tipos de fórmulas

- Decimos que una fórmula del lenguaje de la aritmética es *rudimentaria* si es una fórmula que se obtiene a partir de fórmula atómicas utilizando solamente la negación, conjunción, la disyunción y cuantificaciones acotadas $\forall x < t$ y $\exists x < t$ (donde t puede ser cualquier término del lenguaje que no contenga x).

- Decimos que una fórmula del lenguaje de la aritmética es $\exists$-*rudimentaria* si es una fórmula de la forma $\exists x F$, donde F es rudimentaria, y decimos que una fórmula del lenguaje de la aritmética es $\forall$-*rudimentaria* si es una fórmula de la forma $\forall x F$, donde F es rudimentaria.

- La negación de una fórmula $\exists$-*rudimentaria* es equivalente a una fórmula $\forall$-*rudimentaria* y viceversa.

- Una fórmula $\exists$-*rudimentaria generalizada* es una fórmula que se obtiene a partir de fórmulas rudimentarias utilizando la negación, la conjunción, la disyunción, cuantificaciones acotadas y cuantificaciones existenciales no acotadas.

- Con estas nuevas nociones es posible refinar el lema 16.6:

- **Lema 16.8** Toda función recursiva es aritméticamente definible por medio de una fórmula $\exists$-rudimentaria generalizada.

  Esbozo de prueba Las funciones básicas son definibles a partir de fórmulas $\exists$-rudimentarias generalizadas. Además, mirando la prueba del lemma 16.6 queda claro que sólo hay fórmulas $\exists$-rudimentarias generalizadas involucradas. ∎

- **Proposición 16.10** Toda fórmula $\exists$-rudimentaria generalizada es aritméticamente equivalente a una fórmula $\exists$-rudimentaria.

  Prueba Vea el libro, pp. 205-206. ∎

- De los dos resultados anteriores se sigue que:

- **Lema 16.11** Toda función recursiva es aritméticamente definible *por medio de una fórmula* $\exists$-*rudimentaria.*

Representabilidad

- Decimos que un conjunto S es *definible* en una teoría T si hay una fórmula $F(x)$ tal que:
  - para todo objeto m, si m está en S, entonces $T \vdash F(\mathbf{m})$.
  - para todo objeto m, si m no está en S, entonces $T \vdash \sim F(\mathbf{m})$.
- Definibilidad *aritmética* es simplemente el caso especial en que T es la aritmética verdadera Γ, esto es, el conjunto de todas las oraciones que son verdaderas en el modelo estándar $\mathcal{N}^*$.
- Decimos que una función f es *representable* en una teoría T si hay una fórmula $F(x, y)$ tal que:
  - para todos los objetos m y n, $f(m) = n$ si y sólo si $T \vdash \forall y(F(\mathbf{m}, y) \leftrightarrow y = \mathbf{n})$.
- Esto es equivalente a la conjunción de $T \vdash F(\mathbf{m}, \mathbf{n})$ y $T \vdash \forall y(y \neq \mathbf{n} \rightarrow \sim F(\mathbf{m}, y))$.
- En el caso especial en que T es la aritmética verdadera, representabilidad y definibilidad coinciden.
- Para otras teorías más débiles, sin embargo, la representabilidad es un requisito más fuerte.

Dos teorías aritméticas

- Definimos la teoría $\mathbf{Q}$, llamada 'aritmética mínima', como el conjunto de oraciones del lenguaje de la aritmética que son probables a partir de los siguiente axiomas:

  (Q1) $\forall x(\mathbf{0} \neq x')$
  (Q2) $\forall x \forall y(x' = y' \rightarrow x = y)$
  (Q3) $\forall x(x + \mathbf{0} = x)$
  (Q4) $\forall x \forall y(x + y' = (x + y)')$
  (Q5) $\forall x \forall y(x \cdot \mathbf{0} = \mathbf{0})$
  (Q6) $\forall x \forall y(x \cdot y' = (x \cdot y) + x)$
  (Q7) $\forall x \sim x < \mathbf{0}$
  (Q8) $\forall x \forall y(x < y' \leftrightarrow (x < y \vee x = y))$
  (Q9) $\forall x(\mathbf{0} < x \leftrightarrow (x \neq \mathbf{0}))$
  (Q10) $\forall x \forall y(x' < y \leftrightarrow (x < y \,\&\, x' \neq y))$

- Definimos la teoría $\mathbf{P}$, comunmente llamada 'aritmética de Peano', como el conjunto de oraciones del lenguaje de la aritmética que son probables a partir de (Q1)-(Q-10) junto con el siguiente axioma-esquema:

  (Inducción) $(A(\mathbf{0}) \,\&\, \forall x(A(x) \rightarrow A(x'))) \rightarrow \forall x A(x)$

- Nótese que **Q** es finitamente axiomatizable mientras que **P** no lo es.
- **Q** no es suficientemente fuerte como teoría aritmética, pues no contiene el axioma de inducción, y eso hace que ciertos teoremas aritméticos carezcan de prueba en **Q**. Por ejemplo, $\mathbf{Q} \nvdash \forall x(\mathbf{0} + x = x + \mathbf{0})$.
- Pese a esto, **Q** es todo lo que necesitamos para probar todas las oraciones $\exists$-rudimentarias que son verdaderas en $\mathcal{N}^*$, como el siguiente teorema señala.
- **Teorema 16.13** Una oración $\exists$-rudimentaria es verdadera en $\mathcal{N}^*$ si y sólo si tiene una prueba en **Q**.

  Esbozo de prueba El lado de derecha a izquierda es trivial, pues todos los axiomas de **Q** son verdaderos en $\mathcal{N}^*$ y por ende todos las oraciones $\exists$-rudimentarias que son teoremas de **Q** también son verdaderas en $\mathcal{N}^*$. La conversa es más complicada.

  La estrategia de la prueba consiste en mostrar que para cada tipo de oración $\exists$-rudimentaria F que sea verdadera en $\mathcal{N}^*$, $\mathbf{Q} \vdash F$. Esto puede hacerse llevando a cabo una prueba por inducción matemática. Los detalles pueden encontrarse en el libro (pp. 208-210). ■
- Decimos que una teoría T es ω-incompleta si para cada n, $T \vdash F(\mathbf{n})$ pero $T \nvdash \forall x F(x)$.
- Nótese que en **Q** puede haber fórmulas $\forall$-rudimentarias de la forma $\forall x F(x)$ (donde $F(x)$ es rudimentaria) que son verdaderas en $\mathcal{N}^*$ pero que no tienen prueba en **Q**, aún cuando para cada n, $\mathbf{Q} \vdash F(\mathbf{n})$. Por lo tanto, **Q** es una teoria ω-incompleta (y también lo es **P**).
- Decimos que una función es *rudimentaria* si es aritméticamente definible por medio de una fórmula rudimentaria.
- **Lema 16.14** Toda función rudimentaria es representable en **Q** (y por una fórmula $\exists$-rudimentaria).

  Esbozo de prueba Sea f una función rudimentaria monádica[2]. Ya se ha mostrado que f es definible, de modo que sea $\phi(x, y)$ la fórmula que define aritméticamente f. Ahora consideremos una fórmula $\psi(x, y)$ con la siguiente forma:

  $$\phi(x, y) \,\&\, \forall z < y \sim \phi(x, z)$$

  Nótese que $\psi(x, y)$ es una fórmula rudimentaria. Para concluir que $\psi(x, y)$ representa la función f en **Q** debemos mostrar que:

[2]Esto puede extenderse a funciones de mayor aridad.

1. para todo m y n, $\mathbf{Q} \vdash \psi(\mathbf{m}, \mathbf{n})$ (o, desabreviando: $\mathbf{Q} \vdash \phi(\mathbf{m}, \mathbf{n})$ & $\forall z < \mathbf{n} \sim \phi(\mathbf{m}, z)$); y
2. para todo m y n, $\mathbf{Q} \vdash \forall y(y \neq \mathbf{n} \rightarrow \sim \psi(\mathbf{m}, y))$ (o, desabreviando: $\mathbf{Q} \vdash \forall y(y \neq \mathbf{n} \rightarrow \sim(\phi(\mathbf{m}, y)$ & $\forall z < y \sim \phi(\mathbf{m}, z))))$

Para lo primero, supongamos que $f(m) = n$. Luego, dado que $\phi(x, y)$ define aritméticamente f, tenemos $\mathcal{N}^* \vDash \phi(\mathbf{m}, \mathbf{n})$ y $\mathcal{N}^* \vDash \sim\phi(\mathbf{m}, \mathbf{k})$ para todo $k \neq n$. Luego, también tenemos $\mathcal{N}^* \vDash \forall z < \mathbf{n} \sim\phi(\mathbf{m}, z)$. Por el teorema 16.13, podemos concluir que $\mathbf{Q} \vdash \phi(\mathbf{m}, \mathbf{n})$ & $\forall z < \mathbf{n} \sim\phi(\mathbf{m}, z)$, pues esta fórmula es rudimentaria.

Para lo segundo, nótese que por los axiomas (Q9) y (Q10) se sigue que $\mathbf{Q} \vdash \forall z(\mathbf{m} < z \vee \mathbf{m} = z \vee z < \mathbf{m})$. Esto junto con $\mathbf{Q} \vdash \forall y < \mathbf{n} \sim \phi(\mathbf{m}, y)$ (lo cual ya tenemos por la parte anterior), implica $\mathbf{Q} \vdash \forall y(\phi(\mathbf{m}, y) \rightarrow (y = \mathbf{n} \vee \mathbf{n} < y))$. Ahora bien, de esto último (junto con $\mathbf{Q} \vdash \phi(\mathbf{m}, \mathbf{n})$, lo cual ya fue mostrado) se sigue $\mathbf{Q} \vdash \forall y(\phi(\mathbf{m}, y) \rightarrow y = \mathbf{n} \vee \exists z < y\ \phi(\mathbf{m}, z))$. Pero es sencillo ver que esta última oración es lógicamente equivalente a la oración que buscamos, de modo que $\mathbf{Q} \vdash \forall y(y \neq \mathbf{n} \rightarrow \sim(\phi(\mathbf{m}, y)$ & $\forall z < y \sim \phi(\mathbf{m}, z)))$. ■

- **Lema 16.15** Cualquier composición de funciones rudimentarias es representable en $\mathbf{Q}$ (y por medio de una fórmula $\exists$-rudimentaria).

  Prueba Sean f y g funciones rudimentarias. Por el lema 16.14, estas fórmulas son representables en $\mathbf{Q}$. Supongamos que las fórmulas que las representan son ϕ_f y ϕ_g, respectivamente. Sea $h(x) = g(f(x))$, y consideremos la fórmula ϕ_h:
  $\exists y(\phi_f(x, y)$ & $\phi_g(y, z))$
  Debemos mostrar que ϕ_h representa h en $\mathbf{Q}$. Para cualquier m, sea $f(m) = n$ y sea $k = h(m) = g(f(m))$. Dado que ϕ_f representa f y que $f(m) = n$, se sigue que
  $\mathbf{Q} \vdash \forall y(\phi_f(\mathbf{m}, y) \leftrightarrow y = \mathbf{n})$.
  Además, dado que ϕ_g representa g en $\mathbf{Q}$ y que $g(n) = k$, tenemos:
  $\mathbf{Q} \vdash \forall y(\phi_g(\mathbf{n}, y) \leftrightarrow y = \mathbf{k})$.
  Pero, de estas dos cosas se sigue que:
  $\mathbf{Q} \vdash \forall z(\exists y(\phi_f(\mathbf{m}, y)$ & $\phi_g(y, z)) \leftrightarrow z = \mathbf{k})$.
  En consecuencia, ϕ_h representa h en $\mathbf{Q}$. ■

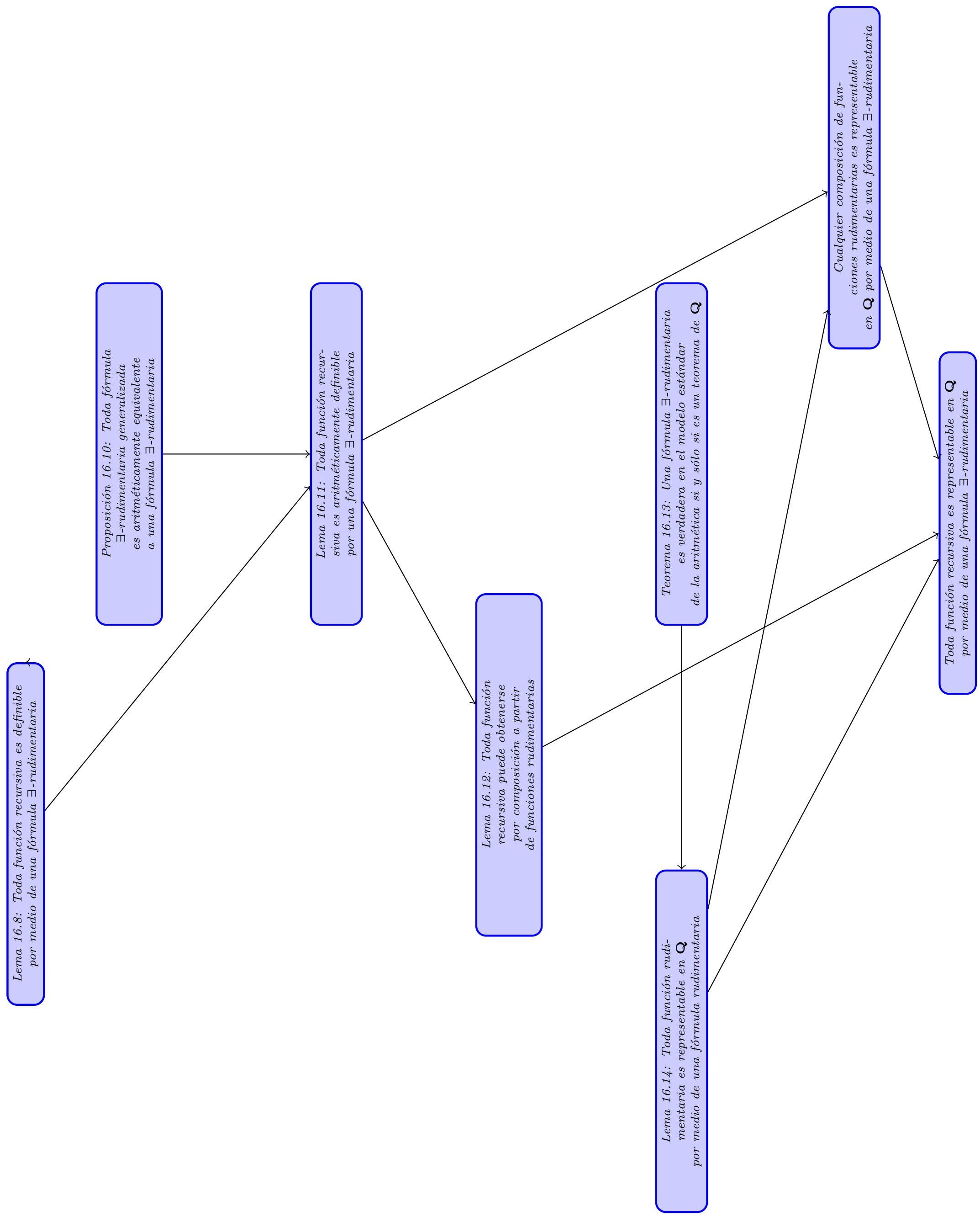
- Podemos usar el lema 16.11 para obtener el siguiente lema:

  Lema 16.12 Toda función recursiva puede obtenerse a partir de una composición de funciones rudimentarias.

  Prueba Vea el libro, p. 206. ■

- **Teorema 16.16** Toda función recursiva es representable en **Q** (y por medio de una fórmula ∃-rudimentaria).

  Prueba Esto se sigue que los lemas 16.12 (cualquier función recursiva puede obtenerse por composición a partir de funciones rudimentarias), 16.14 (toda función rudimentaria es representable en **Q**) y 16.15 (cualquier composición de funciones rudimentarias es representable en **Q**). ■


Lema 16.8: Toda función recursiva es definible por medio de una fórmula ∃-rudimentaria
Proposición 16.10: Toda fórmula ∃-rudimentaria generalizada es aritméticamente equivalente a una fórmula ∃-rudimentaria
Lema 16.11: Toda función recursiva es aritméticamente definible por una fórmula ∃-rudimentaria
Lema 16.12: Toda función recursiva puede obtenerse por composición a partir de funciones rudimentarias
Teorema 16.13: Una fórmula ∃-rudimentaria es verdadera en el modelo estándar de la aritmética si y sólo si es un teorema de Q
Lema 16.14: Toda función rudimentaria es representable en Q por medio de una fórmula rudimentaria
Cualquier composición de funciones rudimentarias es representable en Q por medio de una fórmula ∃-rudimentaria
Toda función recursiva es representable en Q por medio de una fórmula ∃-rudimentaria

Ejercicios

1. Pruebe que la clase de relaciones aritméticamente definibles está cerrada bajo negación, conjunción, disyunción, cuantificación existencial y cuantificación universal, y que toda relación semirecursiva es aritméticamente definible.
2. Utilice los axiomas de la suma para probar que:
   (a) $x + (y + 0)) = (x + y) + 0$
   (b) $x + (y + z) = (x + y) + z \to x + (y + z') = (x + y) + z'$
3. Una teoría T es *inconsistente* si para alguna oración A, tanto A como $\sim A$ son teoremas de T. Una teoría es ω-*inconsistente* si para alguna fórmula $F(x)$, $\exists x F(x)$ es un teorema de T y para cada número n, $\sim F(\mathbf{n})$ es un teorema de T. Sea T una teoría expresada en el lenguaje de la aritmética que extiende a $\mathbf{Q}$. Muestre que:
   (a) Si T prueba cualquier oración $\forall$-rudimentaria que sea falsa en el modelo estándar de la aritmética, entonces T es inconsistente.
   (b) Si T prueba cualquier oración $\exists$-rudimentaria que sea falsa en el modelo estándar de la aritmética, entonces T es ω-inconsistente.
4. Un conjunto P es (positivamente) semidefinible en una teoría T por una fórmula $\phi(x)$ si para todo n, $\phi(\mathbf{n})$ es un teorema de T si y sólo si n está en P. Muestre que todo conjunto semirecursivo es (positivamente) semidefinible en $\mathbf{Q}$ y en cualquier extensión ω-consistente de $\mathbf{Q}$.
5. Considere una interpretación no estándar del lenguaje $\{\mathbf{0}, ', +, \cdot, <\}$ en la cual el dominio es el conjunto de los números naturales junto con un objeto adicional llamado ∞, y donde las relaciones y operaciones sobre los números son las usuales, $\infty' = \infty, x + \infty = \infty + x = \infty$ para todo x, $0 \cdot \infty = \infty \cdot 0 = 0$, pero $x \cdot \infty = \infty \cdot x = \infty$ para todo x, y $x < \infty$ para todo x pero no se da que $\infty < y$ para ningún $y \neq \infty$. Muestre que los axiomas (Q1)-(Q9) son verdaderos en esta interpretación pero el axioma (Q10) es falso.